

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-315197

(43)Date of publication of application : 14.11.2000

(51)Int.Cl. G06F 15/16
G06F 9/46
G06F 13/14

(21)Application number : 2000-084861 (71)Applicant : INTERNATL BUSINESS MACH
CORP <IBM>

(22)Date of filing : 24.03.2000 (72)Inventor : RICHARD BEAROFUSUKI
PATRICK M BRAND

(30)Priority

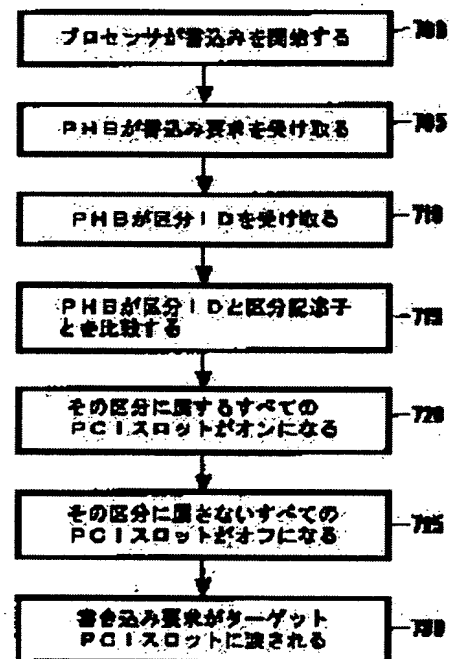
Priority number : 99 283363 Priority date : 31.03.1999 Priority country : US

(54) PCI SLOT CONTROLLER EQUIPPED WITH DYNAMIC CONSTITUTION FOR SECTION SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To obtain a system for operating a multiprocessor computer system by enabling only a communication between an input/output connection where input/output controllers belong to the same process section and a system processor, and dynamically allocating the input/output connection to the process section or deleting it from the process section.

SOLUTION: After a processor starts a writing process (700), a PHB receives a write request (705) and receives the section ID of the start processor (710). Then the PHB compares the section ID with respective section descriptors of its slot (715). The PHB set ON (usable) respective PCI slots having section descriptors showing that it belongs to the same section with the start processor (720) and makes unusable respective PCI slots which do not belong to the section (725). Lastly, the write request is passed to a target PCI device.



LEGAL STATUS

[Date of request for examination] 24.03.2000

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-315197

(P2000-315197A)

(43) 公開日 平成12年11月14日 (2000. 11. 14)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード* (参考)
G 0 6 F 15/16	6 4 0	G 0 6 F 15/16	6 4 0 K
9/46	3 6 0	9/46	3 6 0 C
13/14	3 3 0	13/14	3 3 0 C

審査請求 有 請求項の数17 O L (全 22 頁)

(21) 出願番号 特願2000-84861 (P2000-84861)

(22) 出願日 平成12年3月24日 (2000. 3. 24)

(31) 優先権主張番号 09/283363

(32) 優先日 平成11年3月31日 (1999. 3. 31)

(33) 優先権主張国 米国 (US)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州

アーモンク (番地なし)

(74) 代理人 100086243

弁理士 坂口 博 (外1名)

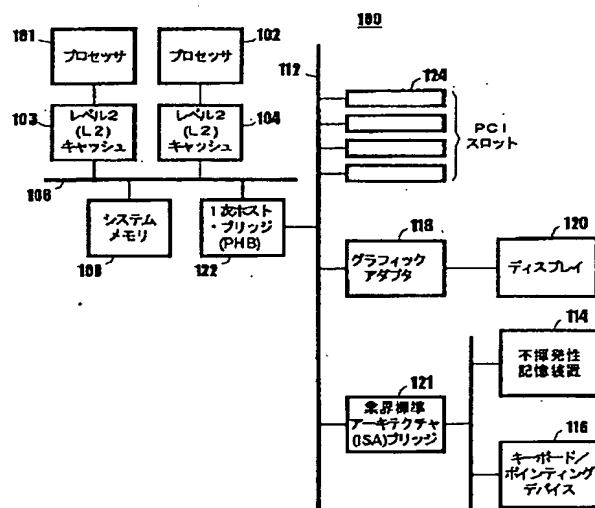
最終頁に続く

(54) 【発明の名称】 区分システム用の動的構成を備えたP C I スロット制御装置

(57) 【要約】

【課題】 区分コンピュータ・システム内の1次ホスト・ブリッジ (P H B) 内の個々のP C I スロットを区分し割り振るためのシステムを提供すること。

【解決手段】 より効率の良いシステム・リソースの割り振りを可能にし、所与の時点で1つまたは複数の区分に1つのP C I スロットを動的に割り振ることができるようにする革新的なP H B システムを含む。



【 特許請求の範囲】

【 請求項1 】 複数の処理区分に分割される複数のシステム・プロセッサであって、各区分が少なくとも1つのシステム・プロセッサと固有の区分記述子とを有する複数のシステム・プロセッサと、
前記プロセッサによって書込みおよび読取りを行うために動作可能に接続される少なくとも1つのメモリと、
前記システム・プロセッサと通信するために接続される入出力コントローラと、
前記入出力コントローラによって管理される複数の入出力接続であって、各入出力接続が少なくとも1つの前記処理区分に割当て可能な複数の入出力接続と、
前記入出力接続に接続される複数の入出力装置とを含み、
前記入出力コントローラが同じ処理区分に属する入出力接続とシステム・プロセッサとの間の通信のみを可能にし、
前記入出力接続が前記処理区分に動的に割り当てられるかまたは前記処理区分から除去することができる、コンピュータ・システム。
【 請求項2 】 所与の処理区分に属するプロセッサが入出力装置と通信しているときに、その区分に属さないすべての入出力接続が前記プロセッサから分離される、請求項1に記載のシステム。
【 請求項3 】 前記分離が電界効果トランジスタを使用して実施される、請求項1に記載のシステム。
【 請求項4 】 前記入出力接続が複数の区分に同時に属することができる、請求項1に記載のシステム。
【 請求項5 】 少なくとも1つのシステム・プロセッサと、
前記システム・プロセッサへの書込みおよび前記システム・プロセッサからの読取りを行うために接続されるメモリと、
前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、
複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置とを含むコンピュータ・システムであって、前記システムが、
前記入出力コントローラにおいて、システム・プロセッサから前記周辺装置に書き込むための要求を受け取るステップと、
前記入出力コントローラにおいて、前記システム・プロセッサに対応する区分IDを受け取るステップと、
前記接続が前記区分IDに対応するグループに属すかどうかに応じて、前記複数の接続の少なくとも1つをオンにするステップと、
前記書込み要求を前記装置に渡すステップとを実行し、
前記入出力接続が前記区分IDに対応する前記グループに動的に割り当てられるかまたは前記グループから除去する

ことができる、コンピュータ・システム。

【 請求項6 】 前記装置接続がPCIスロットである、請求項5に記載のシステム。

【 請求項7 】 前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、請求項5に記載のシステム。

【 請求項8 】 前記グループに属さない前記接続のすべてがオフになる、請求項5に記載のシステム。

【 請求項9 】 前記装置接続が複数の前記グループに同時に属することができる、請求項5に記載のシステム。

【 請求項10 】 複数のシステム・プロセッサと、
前記システム・プロセッサへの書込みおよび前記システム・プロセッサからの読取りを行うために接続される少なくとも1つのメモリと、
前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、
複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置とを含むコンピュータ・システムであって、
前記システムが、
前記入出力コントローラにおいて、前記装置からメモリに書き込むための要求を受け取るステップと、
前記入出力コントローラにおいて、前記装置接続に対応する区分記述子を読み取るステップと、
前記装置接続が前記区分記述子に対応するグループに属すかどうかに応じて、前記装置接続の少なくとも1つをオンにするステップと、
前記グループに属さない前記装置接続のすべてをオフにするステップと、
前記装置からの前記書込み要求を前記メモリに渡すステップとを実行し、
前記装置接続が前記区分IDに対応する前記グループに動的に割り当てられるかまたは前記グループから除去することができる、コンピュータ・システム。

【 請求項11 】 前記装置接続がPCIスロットである、請求項10に記載のシステム。
【 請求項12 】 前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、請求項10に記載のシステム。
【 請求項13 】 前記装置接続が複数の前記グループに同時に属することができる、請求項10に記載のシステム。
【 請求項14 】 複数のシステム・プロセッサと、
前記システム・プロセッサへの書込みおよび前記システム・プロセッサからの読取りを行うために接続される少なくとも1つのメモリと、
前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、
複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置とを含むコンピュータ・システムであって、

【 請求項15 】 前記装置接続がPCIスロットである、請求項10に記載のシステム。

【 請求項16 】 前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、請求項10に記載のシステム。

【 請求項17 】 前記装置接続が複数の前記グループに同時に属することができる、請求項10に記載のシステム。

【 請求項18 】 複数のシステム・プロセッサと、
前記システム・プロセッサへの書込みおよび前記システム・プロセッサからの読取りを行うために接続される少なくとも1つのメモリと、
前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、
複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置とを含むコンピュータ・システムであって、

前記システムが、
前記複数のシステム・プロセッサのそれぞれを1つの処理区分に割り当てるステップと、
前記処理区分のそれぞれにそれぞれの区分IDを割り当てるステップと、
前記装置接続のそれぞれを前記処理区分の少なくとも1つに割り当てるステップと、
メモリにおいて、各装置接続が属す処理区分を識別する情報を記憶するステップと、
同じ処理区分に属す装置接続とシステム・プロセッサとの間で通信を渡し、同じ処理区分に属さない装置接続とシステム・プロセッサとの間で通信を渡さないステップとを実行し、
前記装置接続が前記処理区分に動的に割り当てるかまたは前記処理区分から除去することができる、コンピュータ・システム。

【請求項15】所与の処理区分に属すシステム・プロセッサが同じ処理区分に属す装置接続を介して通信しているときに、その処理区分に属さないすべての装置接続がオフになる、請求項14に記載のシステム。

【請求項16】前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、請求項14に記載のシステム。

【請求項17】前記装置接続が複数の処理区分に同時に属することができる、請求項14に記載のシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、一般にマルチプロセッサ・コンピュータ・システム(partitioned multiprocessor system)に関し、より詳細には区分マルチプロセッサ・システム内のプロセッサ間のリソース割振りに関する。より詳細には、好ましい実施の形態は、マルチプロセッサ・コンピュータ・システム内の個々のPCIスロットを区分し割り振るためのシステムに関する。

【0002】

【従来の技術】マルチプロセッサ・コンピュータ・システムは、当技術分野では周知のものであり、処理タスクを複数の異なるシステム・プロセッサ間で分割できるようにすることにより、処理能力の増加を可能にする。従来のシステムでは、各プロセッサはシステム・リソースのすべてにアクセスすることができ、すなわち、メモリおよび入出力装置など、すべてのシステム・リソースはすべてのシステム・プロセッサ間で共用される。通常、システム・リソースの部品の中には、プロセッサ間で区分できるものもある。たとえば、各プロセッサは共用メモリにアクセスできるようになるが、このメモリは、各プロセッサがそれ自体の作業スペースを有するように分割される。

【0003】より最近では、対称型マルチプロセッサ(SMP)システムは、複数の独立したコンピュータ・

システムとして動作するように区分されていた。たとえば、8つのプロセッサを有する単一システムは、8つのプロセッサのそれぞれ(または1つまたは複数のプロセッサからなる複数のグループ)を処理目的の個別のシステムとして扱うように構成することも可能である。このような「仮想」システムのそれぞれは、それ自体のオペレーティング・システムのコピーを有するはずであり、その場合、独立してタスクが割り当てられる場合もあれば、1つの処理クラスタとして一緒に動作する場合もあり、それにより、高速処理と信頼性の向上の両方が実現される。通常、マルチプロセッサ・システムには、システム構成ならびに特定のプロセッサとの間の共用バスおよび装置上におけるデータの経路指定を含む、システム全体の始動および動作を管理する「サービス」プロセッサも存在する。

【0004】単一マルチプロセッサ・システム内の複数の仮想システムが1つのクラスタとして動作するように構成されているときは、各クラスタ・ノードがマルチプロセッサ内の他の各ノードと通信して、定数折衝および妥当性検査を実行し、「ハートビート」を送信し、いずれかのクラスタ通信技法を使用して他の定数機能を実行できるようにするために、ソフトウェア・サポートを用意しなければならない。これが実施されると、全プロセッサのうちの1つが故障した場合、それによりそのノードがクラスタにとって使用不能な状態になる恐れがあり、そのノードに割り当てられたジョブは、標準のクラスタ技法を使用して、残りのプロセッサ(ノード)間で再割り当てすることができる。

【0005】通常、1つのマルチプロセッサ・システムが複数の仮想システムに分割されると、それぞれの仮想システムはそれ自体のオペレーティング・システムのコピーを有し、各仮想システムに同じオペレーティング・システムが使用される。各プロセッサは同じオペレーティング・システムを実行するので、プロセッサ間のリソース割振りを実現することは比較的容易なことである。

【0006】しかし、現在は、複数の仮想システム間で複数のオペレーティング・システムを実行できる能力を求める市場の要求が存在する。たとえば、ユーザは、ある区分でUNIX(登録商標)変形オペレーティング・システム実行し、第2の区分で「Windows(登録商標)」ベースのオペレーティング・システムを実行したいと希望する場合がある。この必要性はリソース割振りに関連する特定の問題を提起する。すなわち、マルチプロセッサ・システム内の区分間のメモリ分割は一般にハードウェアでサポートされるが、周辺コンポーネント相互接続(PCI)スロットなどの他のリソースの割振りはオペレーティング・システムによって管理される。複数の区分はそれぞれ異なるオペレーティング・システムを実行している可能性があるので、システム・リソースを割り振るための手段であって、オペレーティング・

10

20

30

40

50

システムに基づかないものが必要になる。特に、PCI スロットなどのシステム・リソースをマルチプロセッサ・コンピュータ・システム内の複数区分間で割り振ることができるようにする、オペレーティング・システム独立ソリューションが必要になる。

【0007】

【発明が解決しようとする課題】したがって、本発明の一目的は、マルチプロセッサ・コンピュータ・システムの操作のためのシステムを提供することにある。

【0008】本発明の他の目的は、マルチプロセッサ・コンピュータ・システム内のリソース割振りの改善のためのシステムを提供することにある。

【0009】本発明のさらに他の目的は、マルチプロセッサ・コンピュータ・システム内の個々のPCI スロットを区分し割り振るためのシステムを提供することにある。

【0010】

【課題を解決するための手段】上記の目的は以下に説明するように達成される。区分コンピュータ・システム内の1次ホスト・ブリッジ(PHB)内の個々のPCI スロットを区分し割り振るためのシステムを提供する。より効率の良いシステム・リソースの割振りを可能にし、所与の時点で1つまたは複数の区分に1つのPCI スロットを動的に割り振ることができるようにする革新的なPHBシステムを含む。

【0011】

【発明の実施の形態】次に添付図面、特に図1に関連して説明すると、本発明の好ましい一実施の形態を実現可能なデータ処理システムのブロック図が示されている。データ処理システム100は、たとえば、ニューヨーク州アーモンクのインターナショナル・ビジネス・マシン・コーポレーション(International Business Machine Corporation)から入手可能なパーソナル・コンピュータのデスクトップ・モデルの1つにすることができる。データ処理システム100はプロセッサ101および102を含み、このプロセッサは例示的な実施の形態ではそれぞれレベル2(L2)キャッシュ103および104に接続され、そのキャッシュはシステム・バス106に接続される。

【0012】また、システム・バス106にはシステム・メモリ108および1次ホスト・ブリッジ(PHB)122も接続される。PHB122は入出力バス112をシステム・バス106に結合し、一方のバスからもう一方へデータ・トランザクションを中継するかまたは伝送するかあるいはその両方を行う。例示的な実施の形態では、データ処理システム100は入出力バス112に接続されたグラフィック・アダプタ118を含み、ディスプレイ120用のユーザ・インタフェース情報を受け取る。ハード・ディスク・ドライブにすることができる不揮発性記憶装置114や、従来のマウス、トラックボ

ールなどを含むことができるキーボード/ポインティング・デバイス116などの周辺装置は、業界標準アーキテクチャ(ISA)ブリッジ121を介して入出力バス112に接続される。また、PHB122は入出力バス112を介してPCI スロット124にも接続される。

【0013】図1に示す例示的な実施の形態は単に本発明を説明する目的で示したものであり、当業者であれば、形式と機能の両面で多数の変形形態が可能であることが分かるだろう。たとえば、データ処理システム100は、コンパクト・ディスク読取り専用メモリ(CD-ROM)またはデジタル・ビデオ・ディスク(DVD)ドライブ、サウンド・カードおよびオーディオ・スピーカ、その他の多数の任意選択構成部品も含むことができるだろう。このような変形形態はいずれも、本発明の精神および範囲内に該当すると思われる。データ処理システム100および以下の例示的な区分コンピュータ・システムは、単に説明のための例として示すものであり、アーキテクチャ上の制限を暗示するためのものではない。

【0014】図2を参照すると、本発明の好ましい実施の形態による区分コンピュータ・システムの高レベル・ブロック図が示されている。この図には、3つのマイクロプロセッサ(uP)204~206と、5つのPCI スロット208~212とを有するコンピュータ・システム200が示されている。PCI サブシステム207は、5つのPCI スロットと、PCI ホスト・ブリッジ(PHB)201からなる。この図では、システムは2つの区分に分割され、区分207は、マイクロプロセッサ204および205と、PCI スロット208および209と、PHB201とを含む。また、区分203は、マイクロプロセッサ206と、PCI スロット210~212と、PHB201とを含む。ただし、どちらの区分も同じPHBを共用し、それがPCI スロットの区分を制御することに留意されたい。

【0015】次に図3を参照すると、本発明の好ましい実施の形態による8プロセッサ区分可能コンピュータ・システムのより詳細なブロック図が示されている。この図には、アドレス・バス323、325とデータ・バス324、326のクロスバーを介してプロセッサ301~308に接続されたアドレス・コントローラ321とデータ・コントローラ322とを含むコア・チップセット320を使用して、8つのプロセッサ301~308が対称型マルチプロセッサ構成に接続された状態で示されている。この好ましい実施の形態では、これらのプロセッサはIntel(R)のPentium II(R)クラスのプロセッサである。

【0016】コア論理回路320は、各プロセッサのフロント・サイド・バスとメモリ・サブシステム330、331および入出力バス350とのインタフェースを取り、その入出力バスはアドレス・バス327とデータ・

バス328とを含む。また、コア論理回路は、任意の2つのプロセッサのバス間のスヌープ・トランザクションの量を制限するように設計されたスヌープ・フィルタ329を含む。中央メモリ・サブシステム330、331は、独立して同時にアドレス指定できる2つの個別部分として示されている。

【0017】入出力バス350は、拡張が容易なハイ・パフォーマンス・システムをサポートするために、多数のPCIスロット314~317を付加する能力を提供する。最高4つのPCIホスト・ブリッジ310~313がサポートされ、そのいずれも対等PCIバス・セグメントとのインタフェースを取る。

【0018】好ましい実施の形態では、すべてのPCIスロットはホットプラグ能力を有する。PCIホット・プラグ御論理回路と区分制御論理回路をPHB310~313に統合することにより、好ましい実施の形態ではFETである外部分離回路を使用して、きめの細かい入出力区分を実現するためにトランザクションごとにPCIスロットを分離することができる。各区分は、システム・プロセッサのうちの1つないし全部と、PICスロットのうちの0個ないし全部を所有することができる。

【0019】また、この図には、PHB310を介して接続された様々な入出力装置も示されている。これらは、SCSIコントローラ340と、LAN接続341と、グラフィック・アダプタ342とを含む。ISAブリッジ343は任意のレガシー入出力装置344を接続するために使用され、その入出力装置は、とりわけ、キーボード、マウス、シリアル・ポート、パラレル・ポート、オーディオ装置、フロッピー(登録商標)・ドライブ、CDROM、リアルタイム・クロックを含むことができる。

【0020】図4ないし図6を参照すると、本発明の好ましい実施の形態によるスロットごとのPCI区分のための能力を提供する改良されたPCIホスト・ブリッジが示されている。

【0021】PHB400は、いずれかの方向に同時に発生するアウトバウンド・トランザクションとインバウンド・トランザクションのための別々の要求待ち行列を提供する。アウトバウンド・トランザクションは入出力バス401上のエージェントによって開始され、(8つのプロセッサのうちの1つの代表としての)コア論理回路またはピアツーピアの応用例でPCIメモリ・トランザクションを転送するもう1つのPHBのいずれかからのものであることを意味する。インバウンド・トランザクションはPCIバス403上のエージェントによって開始され、システム・メモリに向けられるかまたはもう1つのPHBの後ろにある対等PCIバス・セグメント上のPCIメモリに向けられる。数ある重要な特徴のうち、PHBは、そのバス・セグメント上のすべてのイニシエータ・エージェントのためのPCIアービトレーシ

ョン、ホット・プラグPCI生挿入/除去のサポート、PCIスロット区分を実現する。

【0022】PHBは、入出力バスから開始されたアウトバウンド・トランザクションに関する通知済み書込みおよび据置き応答と、PCIバスから開始されたインバウンド・トランザクションに関する通知済み書込みおよび据置き応答を実現する。アウトバウンド・メモリ書込みは必ず通知され、入出力書込みは任意選択で通知されるかまたは据置き置かれ、入出力読取りとメモリ読取りはともに据置き応答を呼び出す。据置き応答は、トランザクションがプロセッサの適正順序待ち行列から除去されて、据置きトランザクション待ち行列に入れられ、そのトランザクションが後で完了されることをプロセッサに通知することを意味する。PHB400は、据置きトランザクションをそのアウトバウンド要求待ち行列426に入れ、読取りまたは書込みトランザクションがPCIバス上で完了したときに、PHBは、そのトランザクションが完了したという据置き応答を開始プロセッサに配送することになる。インバウンド・メモリ書込みは必ず通知され、メモリ読取りは遅延される。インバウンド読取りに関する遅延トランザクションは、PHBがそのPCIインバウンド・バッファ429内にトランザクションを入れ、再試行によってPCI開始トランザクションを終了することを意味する。トランザクションは、読取りデータに関する入出力バス401に転送され、PICトランザクションが再試行されるまでインバウンド要求待ち行列428に入れられる。トランザクションが再試行されると、PHBはデータによってトランザクションを完成する。

【0023】入出力バス・トランザクション

8つのシステム・プロセッサのうちの1つから開始された入出力バス・トランザクションは、リソース・アドレス(入出力またはメモリ)と、バス・トランザクション・タイプ情報と、イニシエータ区分IDという側波帯信号450とを含むことになる。このような側波帯信号は、フロント・サイド・バス上のどのプロセッサがPCIをターゲットとするトランザクションを開始したかに基づいて、コア論理回路内で生成され、そのイニシエータがメンバになる区分をPHB400に対して識別する。エージェントを要求するフロント・サイド・バス(FSB)は分散回転優先順位バス・アービトレーション方式を使用するが、この方式はすべてのフロント・サイド・バス・トランザクションのイニシエータを識別するためにたどるものである。各プロセッサには、その区分IDによって定義された区分が割り当てられる。この区分IDは、側波帯信号の生成をサポートするためにコア論理回路に組み込まれた新しいプログラム可能レジスタ・ファイルであり、その信号により本発明の革新的な区分技法が可能になる。

【0024】区分IDレジスタ・ファイルは、システム

初期設定時またはその前に構成されるが、動作中に変更することもできる。コア論理回路は、通常のトランザクションとともに、生成したイニシエータ区分I D側波帯信号を入出力バス上に連結することになる。

【0025】PHB400は、その対等PCIバス・セグメント上のどのPCIスロット・リソースをターゲット区分のメンバとして使用可能にしなければならないかを決定するために、入出力バス・トランザクションによってイニシエータ区分I D側波帯信号を受信する。PHBは、そのPIC構成スペースの一部として、PCIス

ロット区分記述子を含み、それは次のプログラム可能情報を含み、システム初期設定時またはその前に構成される。

【0026】PCIスロット用のターゲット区分I Dターゲット区分I Dは0〜「n」個の区分をサポートし、「n」はシステム内でサポートされるプロセッサの数である。各区分には、PCIスロットのいずれも割り当てないかまたはそのすべてを割り当てることができる。1つのPCIスロットとそのターゲット・リソースのすべてには、所与の時点で1つまたは複数の区分を動的に割り当てることができる。

【0027】PCIスロット用の入出力およびメモリ・アドレス・リソース・デコード・レンジ・レジスタPHBは、これらのレンジ・レジスタを使用して、PCIスロットによってトランザクション・ターゲットとして請求されたアドレス・リソースを肯定的にデコードする。レンジ・レジスタの各セットは、開始(基底)アドレスと終了アドレスとを含むことになるだろう。入出力レジスタはシステム・アドレス・マップからの256バイトの割当て可能範囲という最低限の細分性をサポートし、メモリ・レジスタはシステム・アドレス・マップからの1MBの割当て可能範囲という最低限の細分性をサポートすることになるだろう。PCIスロットごとにサポートされる不連続アドレス・リソースの数としては、最低2つの入出力レンジ・レジスタと2つのメモリ・レンジ・レジスタとの対を必要とする。特定の入出力区分をターゲットとし、入出力バス上で開始されたトランザクションをPHBが受け入れるようにするため、これらのレジスタは肯定的なデコーディングのために使用される。

【0028】PHBが入出力バス開始アウトバンド・トランザクションに関するターゲットとして肯定的に受け入れ、応答する前に、以下の条件が発生しなければならない。以下の条件は、アウトバンド・トランザクション用の肯定区分応答(P・P・R)として参照される。

1. 現行トランザクション用の入出力バス・イニシエータ区分I D側波帯は、PHBによってサポートされるターゲット区分I Dの1つと一致しなければならない。

2. 入出力バス・トランザクション・リソース・アドレス(入出力またはメモリ)は、一致するターゲット区分

に関連する肯定的にデコードされたリソース・ターゲットのアドレス範囲内に該当しなければならない。

【0029】肯定区分応答に関して上記の条件の1つが満たされない場合、PHBは、入出力バス・トランザクションを無視し、入出力バス上に常駐する他のPHBが所期のターゲットであると想定することになる。コア論理回路は、ソフトウェアまたはハードウェアが故障し、それが応答をまったく受信しない場合にバス・トランザクション・ウォッチドッグ・タイマを実現するための要件を有する。バス・トランザクション・タイムアウトなどの場合、コア論理回路は、フロント・サイド・バス上の開始プロセッサにターゲット打切りハード障害応答を返送することになるだろう。

【0030】図4に関連して、肯定区分応答を伴うアウトバンド入出力バス開始トランザクションの場合にPHBから出される様々な応答について以下に要約する。

【0031】アウトバンド・メモリ書込み

メモリ書込みは、アウトバンド要求待ち行列426で通知され、ある項目が使用可能になるとPCIアウトバンド・バッファ427に移動される。トランザクションは、この時点でアウトバンド要求待ち行列426から除去される。PCIイニシエータ/ターゲット制御論理回路425は、トランザクションがPCIアービタ421を介してPCIアウトバンド・バッファ427に入れられたときにPCIバスの所有権を要求し、また、どのPCIスロット・リソースがこの区分のメンバであるかを判定し、それぞれの対応するPCIスロット(複数可)を使用可能にするために、PCIアウトバンド・イニシエータ区分I Dを区分制御論理回路423に転送する。同じPCIバス上の区分内のすべてのスロットは使用可能になる。というのは、一部の装置は、所期のターゲットにならない可能性があるが、そのコンテキストのシャドーイングを行うために特定の装置に関するトランザクションをスヌープするために必要になる可能性がある。PCI区分制御論理回路423は、PCIアウトバンド・イニシエータ区分I Dと、PHBのPCI対等バス・セグメントに付加された各PCIスロット用のターゲット区分I Dとを比較する。PCIアービテーションのためにいかなるサイクルも浪費されず、アービタは現行トランザクション中に次の所有者にバスを授与するが、PCIバスがアイドル状態であることをそれが検出するまで所有権を取得しなくなる。

【0032】現時点で好ましい実施の形態では、PCIバスは、高速バックツーバック・トランザクションを実行するために、競合がまったく発生しないことを保証できる場合、トランザクション間に最低限1つのアイドル・バス状態を必要とすることに留意されたい。PCI規格によって定義される高速バックツーバック・トランザクションとしては2つのタイプがあり、第1のタイプは、現行トランザクション用のマスタのターゲットが直

前の書込みトランザクションのターゲットと同じであるときに実施され、第2のタイプでは、高速バックツープック可能ターゲットがターゲット応答信号上でいっさい競合しないことを保証することが必要になる。PCI規格では、この第2のメカニズムがPCIコマンド・レジスタ内の構成ビットのみによってサポート可能であることが必要である。現時点で好ましい実施の形態では、第2のメカニズムは決して使用可能にならないが、同じ装置に対する第1のメカニズムの高速バックツープック・トランザクションをサポートできることが必要である。

【0033】PCIアイドル・クロックは、あるPCIエージェントが信号の励起を停止し、他のエージェントがその信号の励起を開始するときに競合を避けるためのターンアラウンド・サイクルとして必要になる。好ましい実施の形態では、すべてのバス・トランザクション間でアイドル・バス・クロックを必要とする。というのは、同じ対等PCIバス・セグメントのスロットのきめ細かいPCI区分を実現するために、FET分離スイッチ410～414がこのターンアラウンド・サイクル中に使用可能／使用不能になるからである。区分制御論理回路423は、トランザクションごとにどのPCIスロットを使用可能にすべきかをPCIホット・プラグ制御論理回路に示す。FETスイッチは、PHBがPCIバスの所有権を取得する前のアイドル・ターンアラウンド・クロック・サイクル中に使用可能になる。PHBはメモリ書込みトランザクションを開始し、使用可能になったPCIスロット上のターゲット装置はそのデータを受け取る。PCIターゲットの打ち切りまたはバリエーション・エラーのためにデータが正常にPCIターゲットに配送されない場合、マシン・チェック打ち切り(MCA)が生成される。

【0034】任意選択で通知されるアウトバウンド入出力書込みは、アウトバウンド・メモリ書込みと同じPHB応答を有する。

【0035】アウトバウンド入出力書込みは任意選択で据え置かれる

入出力書込みは、任意選択でアウトバウンド要求待ち行列426に据え置かれ、ある項目が使用可能になるとPCIアウトバウンド・バッファ427に移動される。そのトランザクションは、この時点でアウトバウンド要求待ち行列426から除去されるわけではなく、PCIバス403上で書込みが完了し、PHB400が入出力バス401上に据置き応答トランザクションを配送するまで待ち行列内に存続する。PCIインシエータ／ターゲット制御論理回路425は、トランザクションがPCIアービタ421を介してPCIアウトバウンド・バッファ427に入れられたときにPCIバス403の所有権を要求し、また、どのPCIスロット・リソースがこの区分のメンバであるかを判定し、それぞれの対応するPCIスロット(複数可)440～444を使用可能に

するために、区分制御論理回路423にPCIアウトバウンド・インシエータ区分IDを転送する。PCI区分制御論理回路423は、PCIアウトバウンド・インシエータ区分IDと、PHBのPCI対等バス・セグメントに付加された各PCIスロット440～444用のターゲット区分IDとを比較する。PCIアービタ421は、そのPHBにバスを授与し、PCIバスがアイドル状態であることをそれが検出したときに所有権を取得する。FETスイッチ410～414は、PHB400がPCIバス403の所有権を取得する前のアイドル・ターンアラウンド・クロック・サイクル中に使用可能になる。PHBは入出力書込みトランザクションを開始し、使用可能になったPCIスロット上のターゲット装置はそのデータを受け取る。データが正常にPCIターゲットに配送された場合、据置き応答トランザクションを伴う通常の完了が返される。PCIターゲットの打ち切りまたはバリエーション・エラーのためにデータが正常にPCIターゲットに配送されない場合、据置き応答トランザクションの応答フェーズはハード障害応答を示すことになり、マシン・チェック打ち切り(MCA)が生成される。

【0036】アウトバウンド・メモリまたは入出力読取りは据え置かれる

すべての読取りトランザクションは、アウトバウンド要求待ち行列426に据え置かれ、ある項目が使用可能になるとPCIアウトバウンド・バッファ427に移動される。そのトランザクションは、この時点でアウトバウンド要求待ち行列426から除去されるわけではなく、PCIバス403上で読取りが完了し、PHB400が入出力バス401上に供給された読取りデータを伴う据置き応答トランザクションを出すまで待ち行列内に存続する。PCIインシエータ／ターゲット制御論理回路425は、トランザクションがPCIアービタ421を介してPCIアウトバウンド・バッファ427に入れられたときにPCIバス403の所有権を要求し、また、どのPCIスロット・リソースがこの区分のメンバであるかを判定し、それぞれの対応するPCIスロット(複数可)440～444を使用可能にするために、区分制御論理回路423にPCIアウトバウンド・インシエータ区分IDを転送する。PCI区分制御論理回路423は、PCIアウトバウンド・インシエータ区分IDと、PHBのPCI対等バス・セグメントに付加された各PCIスロット440～444用のターゲット区分IDとを比較する。PCIアービタ421は、そのPHBにバスを授与し、PCIバスがアイドル状態であることをそれが検出したときに所有権を取得する。FETスイッチ410～414は、PHBがPCIバスの所有権を取得する前のアイドル・ターンアラウンド・クロック・サイクル中に使用可能になる。PHBは読取りトランザクションを開始し、使用可能になったPCIスロット上のターゲット装置はそのデータを供給する。データが正常に

PCI ターゲットから読み取られた場合、据置き応答トランザクションおよび読取りデータを伴う通常の完了がデータ・フェーズで供給される。PCI ターゲットの打切りまたはパリティ・エラーのためにデータが正常にPCI ターゲットから読み取られない場合、据置き応答トランザクションの応答フェーズはハード障害応答を示すことになり、マシン・チェック打切り(MCA)が生成される。

【0037】PCI バス開始インバウンド・トランザクションのターゲットとしての肯定区分応答(PPR)は、アウトバウンド・トランザクションの場合よりかなり単純なものである。PCI バス・イニシエータ区分IDは、PHBのPCI アービタ421内で内部生成される。PCI アービタ421は、ポイントツーポイントで、すなわち、中間論理回路または分離FET410～414なしでPCI スロット440～444にそれぞれ接続された信号REQ0:4およびGNT0:4を有する。REQ0:4はスロット440～444用のPCI バス要求線であり、GNT0:4はこれらのスロット用のPCI バス授与信号である。これらの信号はバス化または分離する必要はない。というのは、所与のスロットがオフになり、その他の方法でPHBから分離されたときでも、アービタは要求を受け取り、所有権を授与し、その他の方法で各スロットの状況を検査しなければならないからである。アービタは、現行トランザクション中に次の所有者にバスを授与するが、PCI バスがアイドル状態であることをそれが検出するまで所有権を取得しなくなる。PCI バス・イニシエータ区分IDは、単に、PCI スロット区分記述子に基づいて、どのPCI スロット440～444にバスの所有権を授与すべきか、ならびにどの入出力区分がそのメンバになるかに基づいて生成される。区分制御論理回路423は、PCI バス・イニシエータ区分IDを受け取り、それをPHBによってサポートされるターゲット区分IDのすべてと比較する。同じPCI バス上の1つの区分内のすべてのスロットは、ピアツーピア・トランザクション通信をサポートして使用可能になる。区分制御論理回路423は、どのPCI スロットを使用可能にすべきかをPCI ホット・プラグ制御論理回路に示す。FETスイッチ410～414は、PCI バス・イニシエータがPCI バスの所有権を取得する前のアイドル・クロック・サイクル中に使用可能になる。

【0038】インバウンド・メモリ書込みは通知されるメモリ書込みは、PCI インバウンド・バッファ429で通知され、ある項目が使用可能になるとインバウンド要求待ち行列428に移動される。トランザクションは、この時点でPCI インバウンド・バッファ429から除去される。入出力バスまたはイニシエータ/ターゲット制御論理回路424は、トランザクションがインバウンド要求待ち行列428に入れられたときに入出力バ

ス401の所有権を要求する。バスの所有権が授与され、他のPHB上でシステム・メモリまたはPCIメモリをターゲットとすると、PHBは入出力バス上でメモリ読取りトランザクションを開始する。

【0039】インバウンド・メモリ読取りは遅延されるこの実施の形態のメモリ読取りは、PCI SIG(2575 NE Kathryn St #17Hillsboro, OR 97124)から入手可能であり、参照により本明細書に組み込まれるPCI 2.1規格によって定義される遅延トランザクション・メカニズムによってサポートされる。PHBは、すべてのPCIバス・トランザクション情報をラッチし、インバウンド・メモリ読取りをそのPCIインバウンド・バッファ429に入れ、再試行によってPCIトランザクションを終了する。次にトランザクションは、ある項目が使用可能になるとインバウンド要求待ち行列428に移動され、PCIトランザクションは、この時点でPCIインバウンド・バッファ429から除去される。入出力バス・イニシエータ/ターゲット制御論理回路424は、トランザクションがインバウンド要求待ち行列428に入れられたときに入出力バス401の所有権を要求する。それにバスの所有権が授与されると、PHBは、システム・メモリまたはPCIメモリをターゲットとする入出力バス上でメモリ読取りトランザクションを開始する。読取りデータが返されると、PHBは、インバウンド要求待ち行列428内のデータの首尾一貫性を維持することになる。PCIバス・エージェントが遅延トランザクションと同じメモリ位置を読み取ろうともう一度試みると、PHBはデータで応答するためにPCIインバウンド・バッファ429内にデータを移動し、PCIバス上で遅延トランザクションを完了する。

【0040】システム初期設定および構成
区分コンピュータ・システムを初期設定し構成するための好ましい方法は、システムのサービス・プロセッサが所期区分を確立することである。区分記述子は、最初は基本システム構成中にプログラミングされる。図6に示すように、好ましい方法では、パワーオン・リセットまたはハード・ブート(ステップ610)後に、プロセッサが依然としてリセット状態に保持されている間にサービス・プロセッサがシステム初期設定を開始する(ステップ620)。サービス・プロセッサによって実行されるタスク間で、システムの区分が確立される。それがデフォルト区分構成を使用するように構成されるかどうかに応じて(ステップ630)、サービス・プロセッサは、記憶した情報を使用する(ステップ640)かまたはオペレータから対話的に区分情報を入手する(ステップ650)。サービス・プロセッサは、PHBの位置を含む、システム用の基本メモリ・マップを確立し(ステップ660)、区分記述子レジスタへの区分情報をプログラミングする(ステップ670)。次にシステム・プロセッサは始動され(ステップ680)、各プロセッサ

はその割当て済みPCI スロットを使用してその割当て済み区分内で動作する(ステップ690)。

【0041】区分記述子

図5に示すように、区分記述子は区分メンバシップ情報を含む。このテーブルでは、各スロット項目は、どの区分にそのスロットが属するかを示すために単一ビット・フラグを含む。ここに0:Nとして示す各スロットは、システム内の各区分用の単一ビットを伴う項目を有し、区分の数は一定ではなく、システム・ブート時に設定可能なので、テーブル・サイズは動的に割り振られる。各スロット項目ごとに、各区分に対応するビットは、そのスロットが現在はその区分のメンバではないことを示す「0」か、またはそのスロットが現在はその区分のメンバであることを示す「1」を含む。

【0042】次に図7および図8を参照すると、本発明の好ましい一実施の形態によるサンプルPHBプロセスの簡略流れ図が示されている(より詳細なプロセスは上記の通りである)。図7はプロセッサ開始PCI 書込みを示している。プロセッサが書込みを開始した後(ステップ700)、PHBは書込み要求を受け取り(ステップ705)、開始プロセッサの区分IDを受け取る(ステップ710)。次にPHBは、その区分IDと、そのスロットのそれぞれの区分記述子とを比較する(ステップ715)。PHBは、それが開始プロセッサと同じ区分に属することを示す区分記述子を有する各PCI スロットをオンにし(使用可能にし)(ステップ720)、この区分に属さない各PCI スロットを使用不能にする(ステップ725)。好ましい実施の形態では、これは図4に示すFETを使用することによって行われる。最後に、書込み要求はターゲットPCI 装置に渡される(ステップ730)。

【0043】図8はPCI 装置によって開始されるメモリ 書込みを示している。この図では、PCI 装置が書込みを開始した後(ステップ750)、PHBは要求を受け取り(ステップ755)、それがどの区分に属するかを決定するためにそのPCI スロットの区分記述子を読み取る(ステップ760)。それがこのように実行した後、PHBは、その区分に属すすべてのスロットを使用可能にし(ステップ765)、その区分に属さないすべてのスロットを使用不能にする(ステップ770)。最後に、書込み要求はPHBによって入出力バスに渡される(ステップ755)。

【0044】動的スロット 割当て

PCI スロットは、動的に割り当てることもでき、複数の所有者を有することもできる。以下の説明では、排他スロット 所有権と複数所有権の両方の場合に、1つの区分からの複数のスロットを動的に割り当て、除去するためのプロセスについて詳述する。排他所有権とは、ある時点に1つのスロットを唯一の区分が所有するが、ランタイム中にそのスロットを他の区分に動的に移動できる

場合である。複数所有権とは、ある時点に1つのスロットを複数の区分が所有することができ、ランタイム中に区分所有権から1つのスロットを割り当てたり、割当て解除することができる場合である。

【0045】除去-排他所有権

次に図9を参照すると、それがもはや特定のスロットの所有権を必要としないある区分が判断したときに(ステップ800)、オペレーティング・システムはその区分から除去するためのスロットを選択する(ステップ805)。次にオペレーティング・システムは、入出力バス上の変更の開始を通知するホットプラグ事象を開始する(ステップ810)。次にオペレーティング・システムは、区分所有権から除去すべきスロット内の装置を静止する(ステップ815)。このステップは、すべての保留作業を完了することと、新しい作業がその装置に対して行われるのを防止することを含む。そのスロット内の装置がすべての保留活動を完了すると、オペレーティング・システムは、他のどの装置ももはや対応するデバイス・ドライバを必要としない場合に、そのデバイス・ドライバをアンロードすることを選択できる。次にオペレーティング・システムは、PCI 構成レジスタを介してそのスロット内の装置を使用不能にする(ステップ820)。このコンテキストでは、PCI 使用不能とは、その装置がバス活動に関与しないようにPCI コマンド・レジスタ・ビットをプログラミングすることを意味する。装置を使用不能にすることは、スロットが将来の任意の時点に再活動化される場合にその装置が非活動状態になることを保証するための安全機能である。最後に、オペレーティング・システムは、そのスロットに対する区分の所有権を除去するように対応するスロット 区分記述子をプログラミングする(ステップ825)。これで除去動作を完了する(ステップ830)。

【0046】オペレーティング・システムが実行するアクションに関してこのプロセスを説明するが、これらの機能は、問題の区分内で実行される他のソフトウェアによって実行することができる。これは、その動作を調整し、装置を実施することができる適切なソフトウェアを含むはずである。

【0047】追加-排他的所有権

次に図10を参照すると、排他的に所有されるスロットをある区分に追加するためのプロセスが示されている。その区分がスロットを必要とするときに(ステップ835)、その区分内に含めるべきスロットが選択される(ステップ840)。オペレーティング・システムは、そのスロットが現在所有されているかどうかまたはそのスロットが所有されていないかどうかを判定するために検査する(ステップ845)。この検査は、ターゲット・スロット用のスロット 記述子を読み取ることによって実行される。そのスロットが所有されている場合、そのスロットは区分に含めるために使用可能ではなく、した

がって、「エラー・スロット 使用不能」などのエラーが表示される（ステップ850）。そのスロットが現在所有されていない（使用可能である）場合、オペレーティング・システムは、任意選択でそのスロット上でハード・リセットを実行することを選択することができる（ステップ855）。ハード・リセットを実行することは、スロット内の装置がリセットされ、非活動状態であることを保証するためにオペレーティング・システムが行える方針上の選択である。次にオペレーティング・システムは、そのスロットを区分に含むように対応するスロット区分記述子をプログラミングする（ステップ860）。そのスロットが区分内に入ると、そのスロット内の装置はオペレーティング・システムにとって可視状態になる。

【0048】次にオペレーティング・システムは、入出力バス上の変更の開始を通知するホット・プラグ事象を開始する（ステップ865）。オペレーティング・システム（または、より具体的にはオペレーティング・システム内のPCIバス・ドライバ）は、アダプタと、潜在的にバスを構成する（ステップ870）。このコンテキストでは、構成プロセスは、非競合（固有の）リソース（たとえば、メモリ・スペース）をその装置に割り当てることを含む。バスを構成することは、バス・ドライバがバス全体を検査して構成パラメータを決定し、新しい装置の追加に対処するためにおそらく他の装置内のリソースを再割当てすることを意味する。

【0049】次に装置は構成されてオンライン状態になり、オペレーティング・システムは対応するデバイス・ドライバをロードするかまたはその装置が追加され、使用可能であることをすでにロードしたドライバに通知する（ステップ875）。これで追加動作を完了する（ステップ880）。

【0050】除去-複数所有者

次に図11を参照すると、それがもはや特定のスロットの所有権を必要としないある区分が判断したときに（ステップ900）、それはその区分から除去するためのスロットを指定する（ステップ905）。オペレーティング・システムは、入出力バス上の変更の開始を通知するホットプラグ事象を開始する（ステップ910）。次にオペレーティング・システムは、区分所有権から除去すべきスロット内の装置を静止する（ステップ915）。このステップは、すべての保留作業を完了することと、新しい作業がその装置に対して行われるのを防止することを含む。

【0051】そのスロット内の装置がすべての保留活動を完了すると、オペレーティング・システムは、他のどの装置ももはや対応するデバイス・ドライバを必要としない場合に、そのデバイス・ドライバをアンロードすることを選択できる。そのスロットが複数所有されているかどうかを判定するために検査が行われる（ステップ9

20）。そのスロットが複数所有されていない場合、オペレーティング・システムはPCI構成レジスタを介してそのスロット内の装置を使用不能にする（ステップ935）。この場合、「PCI使用不能」とは、その装置がバス活動に関与しないようにPCIコマンド・レジスタ・ビットをプログラミングすることを意味する。装置を使用不能にすることは、スロットが将来の任意の時点に再活動化される場合に非活動装置を保証するための安全機能である。そのスロットが複数所有されている場合、その装置は他の区分内でも活動状態になっているので、オペレーティング・システムはその装置を使用不能にすることができない。

【0052】最後に、オペレーティング・システムは、そのスロットに対する区分の所有権を除去するように対応するスロット区分記述子をプログラミングする（ステップ925）。これで除去動作を完了する（ステップ930）。

【0053】追加-複数所有者

次に図12および図13を参照すると、これは最も複雑な事例である。というのは、割当て済みリソースを伴う現在活動状態の装置がバス上に導入されるからである（ステップ1000）。ある区分のバス・セグメント内に活動装置を含めるには、非競合構成を確立しなければならない。非競合装置がバス上に導入された場合、エラーが発生することになる。

【0054】その区分内に含めるべきスロットが選択される（ステップ1002）。オペレーティング・システムは、そのスロットが現在所有されているかどうかまたはそのスロットが所有されていないかどうかを判定するために検査する（ステップ1004）。この検査は、ターゲット・スロット用のスロット記述子を読み取ることによって実行される。そのスロットが所有されている場合、プロセスは、図13に関連して以下に説明する複数所有者解明に移行する。

【0055】そのスロットが現在所有されていない（使用可能である）場合、オペレーティング・システムは、任意選択でそのスロット上でハード・リセットを実行することを選択することができる（ステップ1006）。ハード・リセットを実行することは、そのカードがリセットされ、非活動状態であることを保証するためにオペレーティング・システムが行える方針上の選択である。次にオペレーティング・システムは、そのスロットを区分に含むように対応するスロット区分記述子をプログラミングする（ステップ1008）。そのスロットが区分内に入ると、そのスロット内の装置はオペレーティング・システムにとって可視状態になる。次にオペレーティング・システムは、入出力バス上の変更の開始を通知するホット・プラグ事象を開始する（ステップ1010）。複数所有権（すなわち、現行区分と他の区分の両方で活動状態である装置）の複雑さが追加されるので、

装置およびバス構成は可視状態の(この区分がアクセスできるスロット)複数所有装置を特別に考慮に入れなければならない。オペレーティング・システムは、排他的に所有される装置の構成レジスタに対してのみ変更を行えることを把握して、その装置を構成しようと試みる(ステップ1012)。他の装置は固定かつ不変なものとして扱われる。

【0056】適当な構成を達成できる場合、装置は構成されてオンライン状態になり、オペレーティング・システムは対応するデバイス・ドライバをロードするかまたはその装置が追加され、使用可能であることをすでにロードしたドライバに通知する(ステップ1014)。これで追加動作を完了する(ステップ1016)。

【0057】適当な構成を達成しなかった場合、単純な事例では動作が失敗に終わるはずである。しかし、高度のシナリオでは、適当で相互に合致した装置構成値のセットを確立するために、すべての所有区分間(およびその中で実行されているオペレーティング・システム間)で連携努力が行われる(ステップ1018)。連携する際にオペレーティング・システムが適当な装置構成値のセットに到達できる場合、これらの値が適用される(ステップ1020)。オペレーティング・システムは前の通り、続けてドライバをロードするかまたはドライバに通知し(ステップ1014)、動作が完了する(ステップ1016)。

【0058】いかなる連携構成も達成できない場合、オペレーティング・システムは、区分記述子をプログラミングすることにより、その区分からそのスロットを除去し(ステップ1022)、そのスロットを追加できなかったというエラーを通知しなければならない(ステップ1024)。

【0059】適当なバス構成を決定するためのオペレーティング・システム間の連携努力では、連続する区分が装置の所有権をアサート解除し、受入れ可能な構成を入手するかまたはアサート解除プロセスを使い果たすまでバス構成を再試行する必要がある。

【0060】次に図13を参照すると、このプロセスは、あるスロットが他の区分によってすでに所有されている場合にステップ1026に到達し、そのスロット内の装置がすでに構成され、動作可能であることを意味する。

【0061】オペレーティング・システムは、追加すべきスロット内の装置が構成値(すなわち、非競合リソース割当て)の互換セットを有するかどうかを判定するために検査する(ステップ1026)。この検査は、構成データ用の所有区分の1つ(またはそこで実行されるオペレーティング・システム)を尋ねるかまたはサービス・プロセッサなどの中央設置場所から構成データを検索することにより実施される。この選択は実施態様に依存し、当業者の能力の範囲内である。

【0062】構成値が互換性のあるものである場合、オペレーティング・システムは、そのスロットを区分内に含むように対応するスロット区分記述子を実行する(ステップ1028)。そのスロットが区分内にあると、そのスロット内のどの装置もオペレーティング・システムにとって可視状態になる。次にオペレーティング・システムは、入出力バス上の変更の開始を通知するホット・プラグ事象を開始する(ステップ1030)。装置は構成されてオンライン状態になり、オペレーティング・システムは対応するデバイス・ドライバをロードするかまたはその装置が追加され、使用可能であることをすでにロードしたドライバに通知する(ステップ1032)。これで追加動作を完了する(ステップ1042)。

【0063】追加すべきスロット内の装置が互換構成値を有していない場合、オペレーティング・システムは、複数所有装置の「周り」のバスを構成するよう試みる(ステップ1034)。すなわち、複数所有装置構成データを固定かつ不変なものとして保持しながら、その構成を再計算する。この再計算が成功した場合、新しい構成値が適用され(ステップ1036)、プロセスはステップ1028から動作を再開する。再計算が成功しない場合、動作は単に失敗に終わるだけである。

【0064】高度な事例では、ステップ1018のものと同様の「連携再構成」が試行される(ステップ1038)。連携再構成が成功した場合、その構成が適用され、プロセスが再開する(ステップ1036)。

【0065】連携プロセスが失敗した場合、適当な構成はまったく入手できず、スロット追加は失敗し、それに応じてエラーが報告される(ステップ1040)。

【0066】ただし、この複雑な事例では、適当な構成が決定された後でのみ、活動カードがその区分に含まれることに留意されたい。プロセス内の早い時期に区分記述子をプログラミングすると、その結果、バス上の装置間でリソース競合が発生する可能性もある。

【0067】複数所有権の考慮事項

ある装置が物理的レベルで複数の所有者を有する場合、特別なプログラミングの考慮事項が必要である。対比のため、論理的レベルで1つの装置を共用する例としては共用ネットワーク・プリンタがある。この共用プリンタの場合、そのプリンタの物理的所有者は依然として一人だけであり、システムは実際にはパラレル・ポート・ケーブルを介してプリンタに配線される。装置の物理的操作を管理し制御するのはこのサーバ・システム(プリンタにケーブル配線されたもの)である。

【0068】論理的共用とは対照的に、物理的共用では、所有するオペレーティング・システム間の追加調整が必要である。オペレーティング・システムは、物理的に共用される装置へのアクセスが個別ユニットで行われることを保証しなければならない。個別動作は、共用装置の性質に応じて様々である。前述のプリンタの例の場合

10

20

30

40

50

合、1つの個別動作は完全なプリント・ジョブである。活動プリント・ジョブは、第2のジョブが装置に導入される前に完了できるようにしなければならない。この個別プロトコルに従わない場合、両方のプリント・ジョブは破壊された状態になる。オペレーティング・システムは、当業者の能力の範囲内で考慮される「トークン・パッシング」または共用セマフォを含む個別動作中に排他所有権を保証するために任意の標準手段を選択することができる。

【0069】修正形態および変形形態

当然のことながら、本発明の精神および範囲から逸脱せずに開示されたシステムおよび方法に対してなすことが可能な多くの修正形態および変形形態が存在する。たとえば、上記の説明では特に周辺コンポーネント相互接続(PCI)接続の割振りを論じているが、スロット接続の排他区分および選択的分離の技法を含む、これらの技法は多数の異なるコンピュータ・アーキテクチャおよびシステムに適用することができる。

【0070】さらに、代替実施形態では、各スロットごとに単一PHBを使用することができる。したがって、既存の装置の周りに構成する必要性を除去する装置(すなわち、スロット)はバス・セグメントあたり1つだけであるので、より複雑な構成問題の一部は検出されない。形式および詳細における他の変形形態は確かに当業者の能力の範囲内であり、特許請求の範囲の範囲内に該当すると予想される。

【0071】まとめとして、本発明の構成に関して以下の事項を開示する。

【0072】(1) 複数の処理区分に分割される複数のシステム・プロセッサであって、各区分が少なくとも1つのシステム・プロセッサと固有の区分記述子とを有する複数のシステム・プロセッサと、前記プロセッサによって書き込みおよび読取りを行うために動作可能に接続される少なくとも1つのメモリと、前記システム・プロセッサと通信するために接続される入出力コントローラと、前記入出力コントローラによって管理される複数の入出力接続であって、各入出力接続が少なくとも1つの前記処理区分に割当て可能な複数の入出力接続と、前記入出力接続に接続される複数の入出力装置とを含み、前記入出力コントローラが同じ処理区分に属する入出力接続とシステム・プロセッサとの間の通信のみを可能にし、前記入出力接続が前記処理区分に動的に割り当てるかまたは前記処理区分から除去することができる、コンピュータ・システム。

(2) 所与の処理区分に属するプロセッサが入出力装置と通信しているときに、その区分に属さないすべての入出力接続が前記プロセッサから分離される、上記(1)に記載のシステム。

(3) 前記分離が電界効果トランジスタを使用して実施される、上記(1)に記載のシステム。

(4) 前記入出力接続が複数の区分に同時に属することができる、上記(1)に記載のシステム。

(5) 少なくとも1つのシステム・プロセッサと、前記システム・プロセッサへの書き込みおよび前記システム・プロセッサからの読取りを行うために接続されるメモリと、前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置とを含むコンピュータ・システムであって、前記システムが、前記入出力コントローラにおいて、システム・プロセッサから前記周辺装置に書き込むための要求を受け取るステップと、前記入出力コントローラにおいて、前記システム・プロセッサに対応する区分IDを受け取るステップと、前記接続が前記区分IDに対応するグループに属すかどうかに応じて、前記複数の接続の少なくとも1つをオンにするステップと、前記書き込み要求を前記装置に渡すステップとを実行し、前記入出力接続が前記区分IDに対応する前記グループに動的に割り当てるかまたは前記グループから除去することができる、コンピュータ・システム。

(6) 前記装置接続がPCIスロットである、上記(5)に記載のシステム。

(7) 前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、上記(5)に記載のシステム。

(8) 前記グループに属さない前記接続のすべてがオフになる、上記(5)に記載のシステム。

(9) 前記装置接続が複数の前記グループに同時に属することができる、上記(5)に記載のシステム。

(10) 複数のシステム・プロセッサと、前記システム・プロセッサへの書き込みおよび前記システム・プロセッサからの読取りを行うために接続される少なくとも1つのメモリと、前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置とを含むコンピュータ・システムであって、前記システムが、前記入出力コントローラにおいて、前記装置からメモリに書き込むための要求を受け取るステップと、前記入出力コントローラにおいて、前記装置接続に対応する区分記述子を読み取るステップと、前記装置接続が前記区分記述子に対応するグループに属すかどうかに応じて、前記装置接続の少なくとも1つをオンにするステップと、前記グループに属さない前記装置接続のすべてをオフにするステップと、前記装置からの前記書き込み要求を前記メモリに渡すステップとを実行し、前記装置接続が前記区分IDに対応する前記グループに動的に割り当てるかまたは前記グループから除去することができる、コンピュータ・システム。

(11) 前記装置接続がPCIスロットである、上記

(1 0) に記載のシステム。

(1 2) 前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、上記(1 0) に記載のシステム。

(1 3) 前記装置接続が複数の前記グループに同時に属することができる、上記(1 0) に記載のシステム。

(1 4) 複数のシステム・プロセッサと、前記システム・プロセッサへの書込みおよび前記システム・プロセッサからの読取りを行うために接続される少なくとも1つのメモリと、前記メモリおよび前記プロセッサと通信するために接続される入出力コントローラと、複数の装置接続の1つを介して前記入出力コントローラと通信するために接続される少なくとも1つの周辺装置を含むコンピュータ・システムであって、前記システムが、前記複数のシステム・プロセッサのそれぞれを1つの処理区分に割り当てるステップと、前記処理区分のそれぞれにそれぞれの区分IDを割り当てるステップと、前記装置接続のそれぞれを前記処理区分の少なくとも1つに割り当てるステップと、メモリにおいて、各装置接続が属す処理区分を識別する情報を記憶するステップと、同じ処理区分に属す装置接続とシステム・プロセッサとの間で通信を渡し、同じ処理区分に属さない装置接続とシステム・プロセッサとの間で通信を渡さないステップとを実行し、前記装置接続が前記処理区分に動的に割り当てるかまたは前記処理区分から除去することができる、コンピュータ・システム。

(1 5) 所与の処理区分に属すシステム・プロセッサが同じ処理区分に属す装置接続を介して通信しているときに、その処理区分に属さないすべての装置接続がオフになる、上記(1 4) に記載のシステム。

(1 6) 前記装置接続がオフになったときに、それらが電界効果トランジスタによって前記入出力コントローラから分離される、上記(1 4) に記載のシステム。

(1 7) 前記装置接続が複数の処理区分に同時に属することができる、上記(1 4) に記載のシステム。

【図面の簡単な説明】

【図1】本発明の好ましい一実施の形態による例示的なコンピュータ・システムのブロック図である。

【図2】本発明の好ましい一実施の形態による区分コンピュータ・システムの高レベル・ブロック図である。

【図3】本発明の好ましい一実施の形態による8プロセッサ・コンピュータ・システムのより詳細なブロック図である。

【図4】本発明の好ましい一実施の形態による改良されたPCIホスト・ブリッジを示す図である。

【図5】本発明の好ましい一実施の形態による区分記述子テーブルである。

【図6】本発明の好ましい一実施の形態によるシステム構成プロセスの流れ図である。

10 【図7】本発明の好ましい一実施の形態によるプロセッサ開始PCI書込みを示す図である。

【図8】本発明の好ましい一実施の形態によりPCIデバイスによって開始されるメモリ書込みを示す図である。

【図9】本発明の好ましい一実施の形態による排他所有権スロット除去動作を示す図である。

【図10】本発明の好ましい一実施の形態による排他所有権スロット追加動作を示す図である。

20 【図11】本発明の好ましい一実施の形態による複数所有権スロット除去動作を示す図である。

【図12】本発明の好ましい一実施の形態による複数所有権スロット追加動作を示す図である。

【図13】本発明の好ましい一実施の形態による複数所有権スロット追加動作を示す図である。

【符号の説明】

100 データ処理システム

101 プロセッサ

102 プロセッサ

103 レベル2(L2)キャッシュ

30 104 レベル2(L2)キャッシュ

106 システム・バス

108 システム・メモリ

112 入出力バス

114 不揮発性記憶装置

116 キーボード/ポインティング・デバイス

118 グラフィック・アダプタ

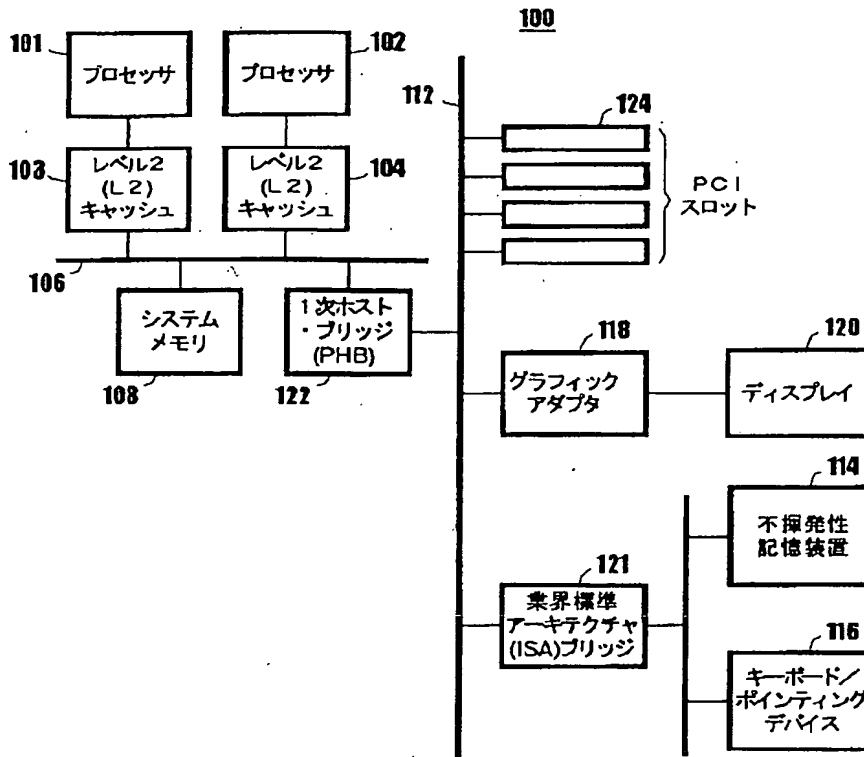
120 ディスプレイ

121 業界標準アーキテクチャ(ISA)ブリッジ

122 1次ホスト・ブリッジ(PHB)

40 124 PCIスロット

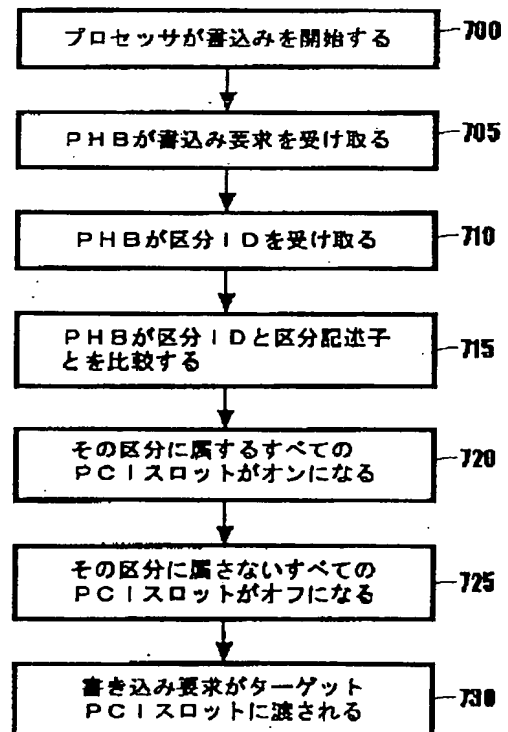
【 図1 】



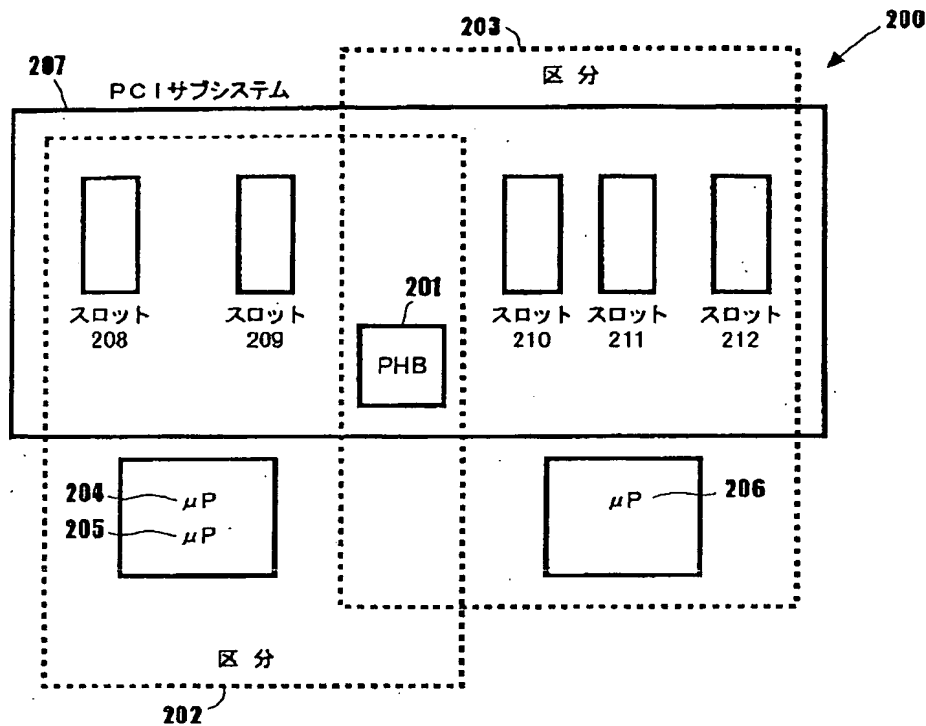
【 図5 】

	ビット n	区分記述子	ビット 1	ビット 0
スロット 0	0=no 1=yes		0=no 1=yes	0=no 1=yes
スロット 1				
動的 拡張リ		ビットn「=」区分n		
スロット n				
	↑ 区分 n		↑ 区分 1	↑ 区分 0

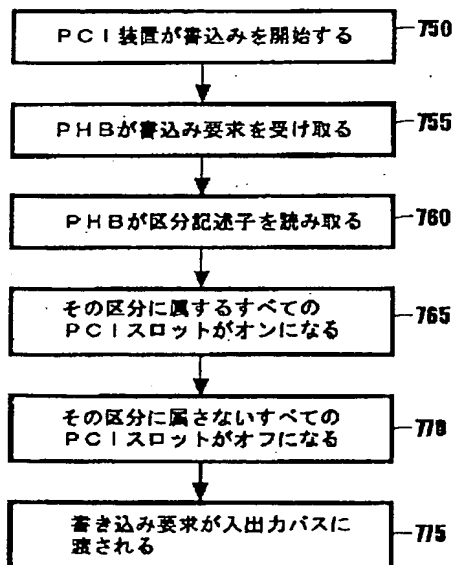
【 図7 】



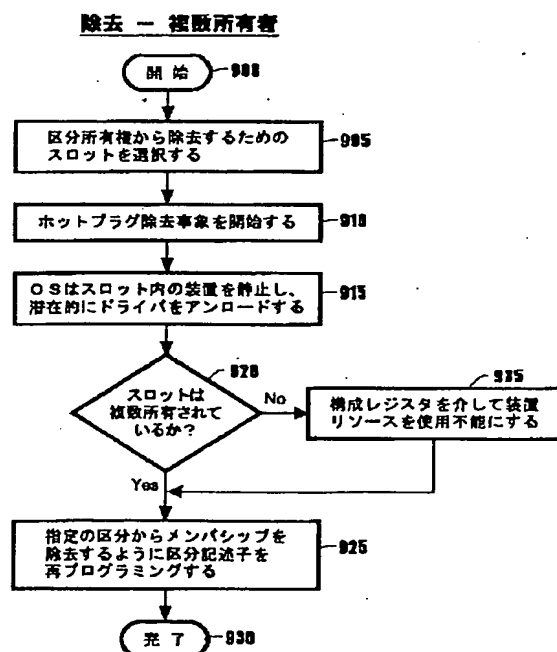
【 図2 】



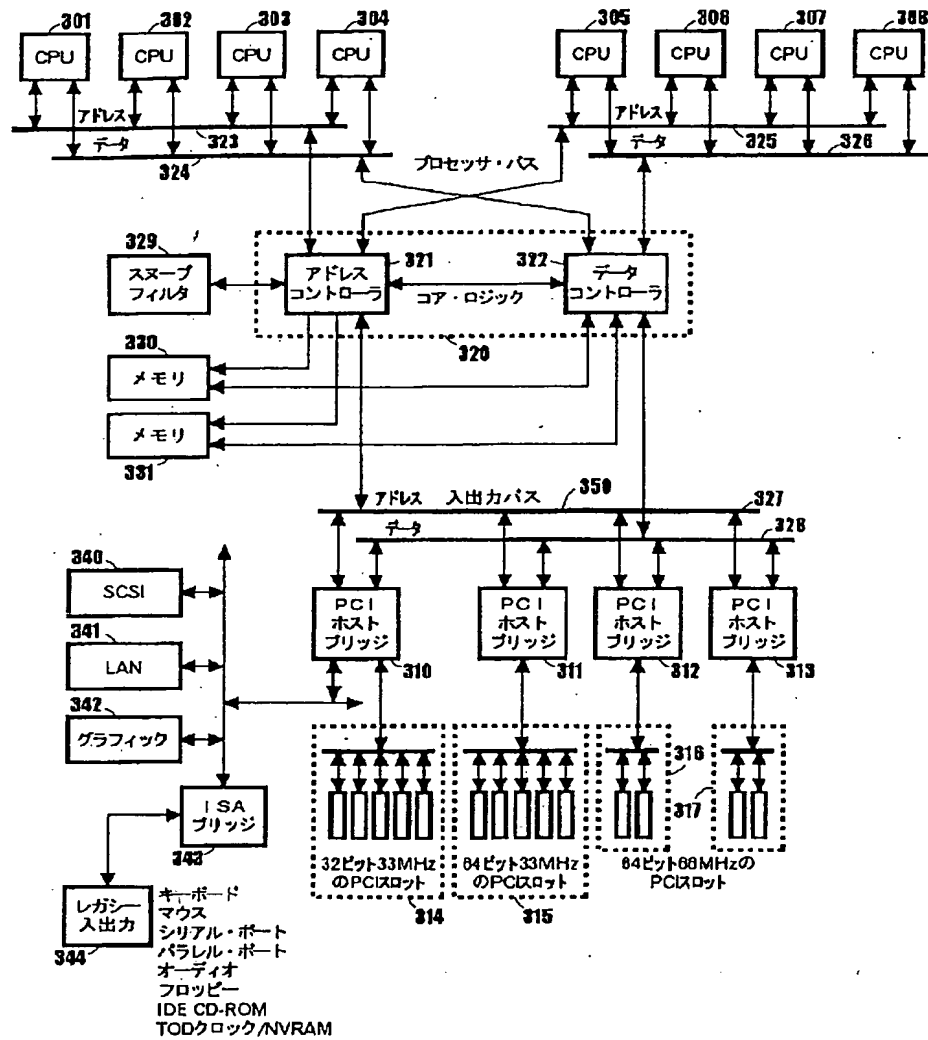
【 図8 】



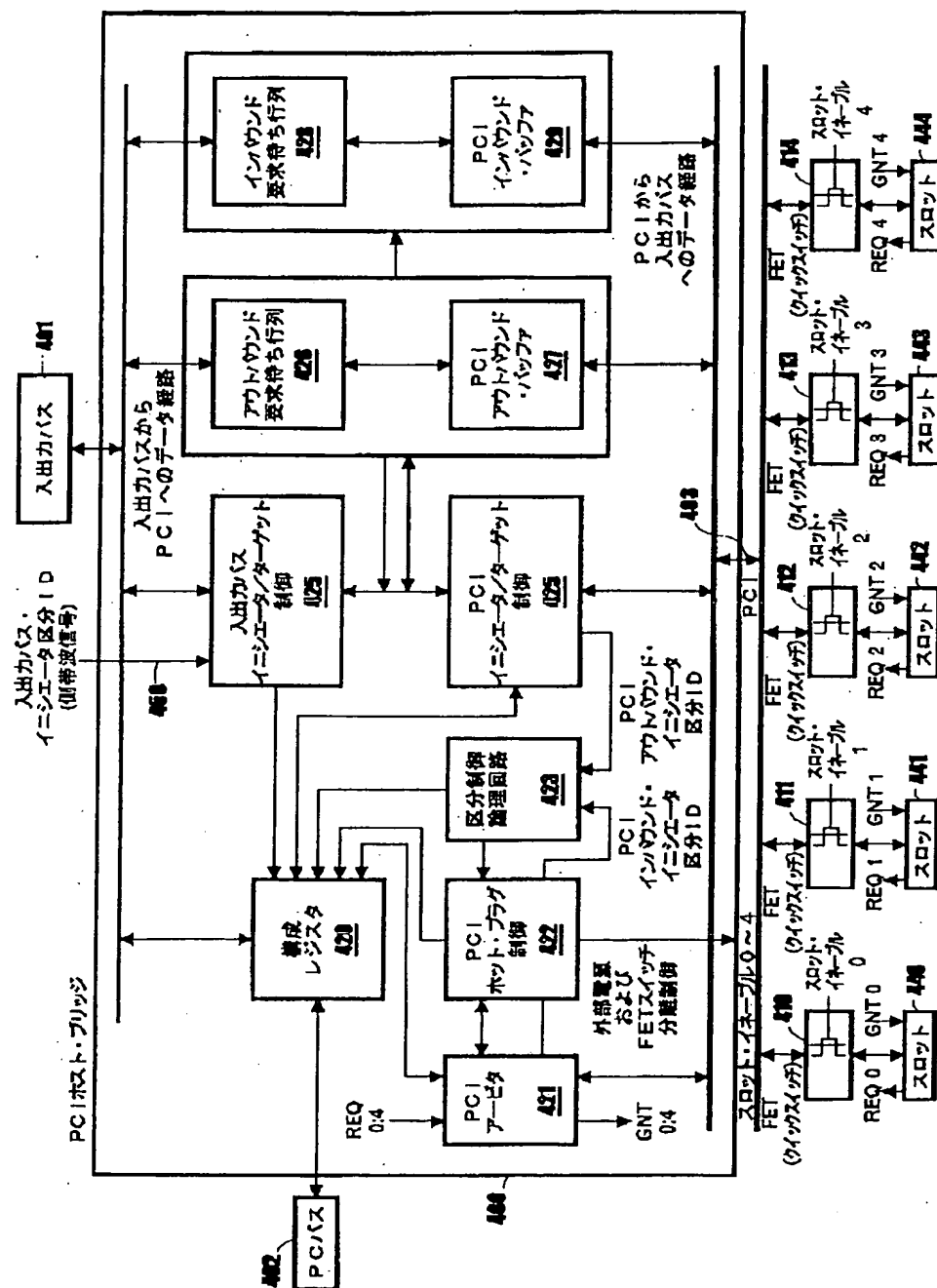
【 図11 】



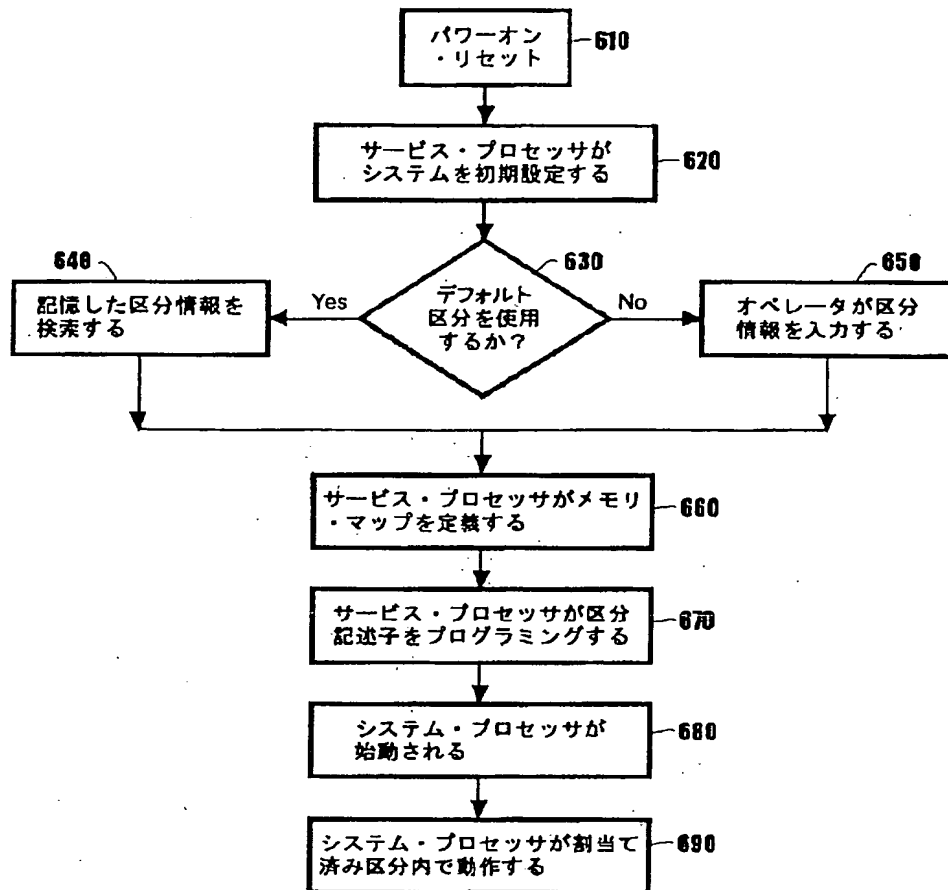
【 図3 】



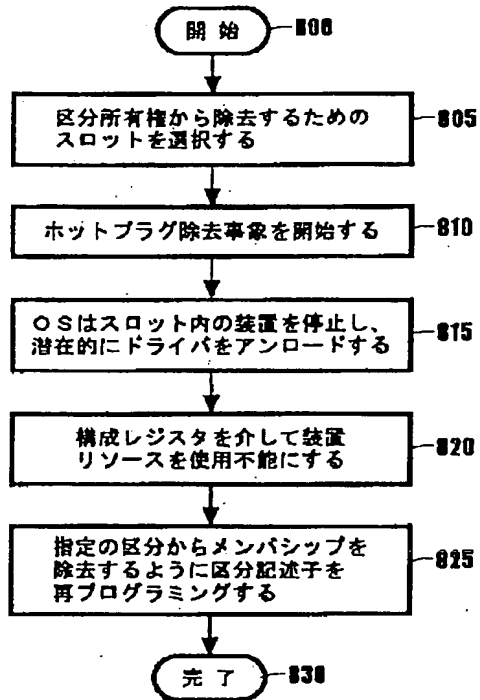
【 図4 】



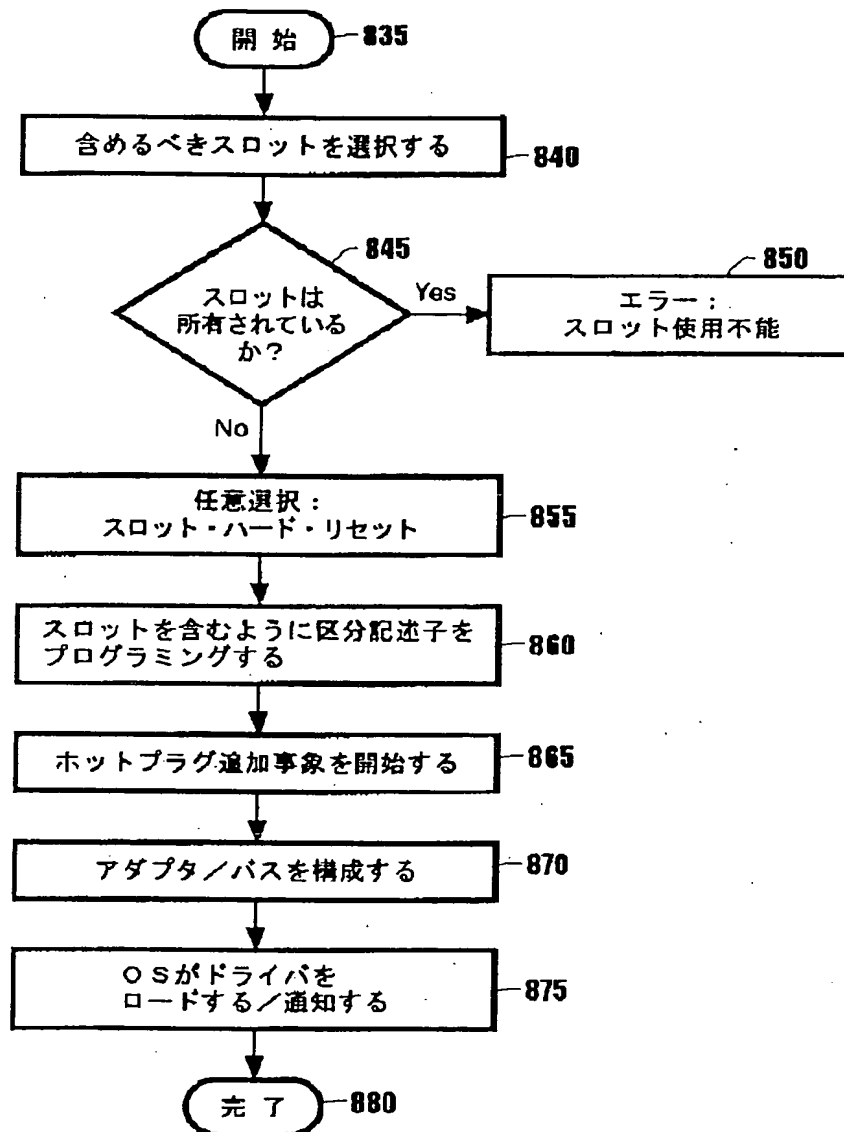
【 図6 】



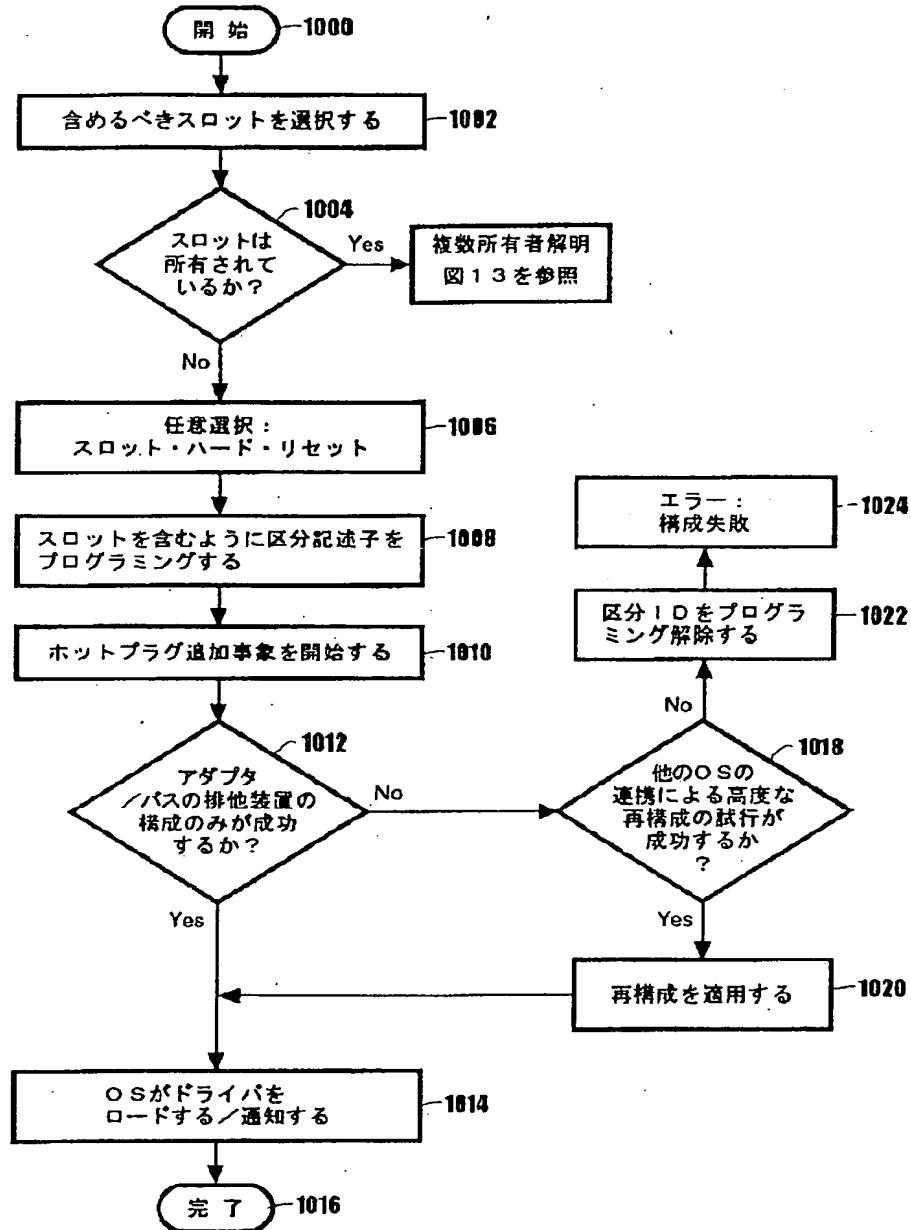
【 図9 】

除去 — 排他所有権

【 図10 】

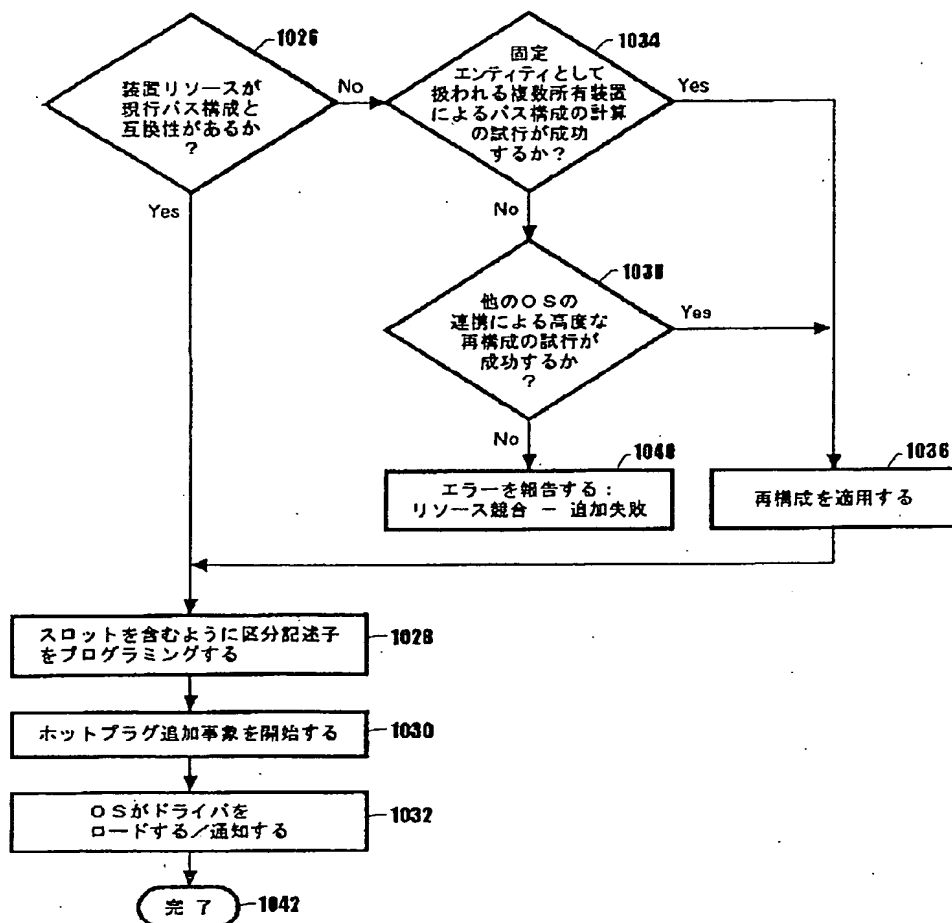
追加 — 排他所有権

【 図12 】

追加 — 複数所有者

【 図13 】

追加 — 複数所有者解明



フロント ページの続き

(72)発明者 リチャード・ベアロフスキ
 アメリカ合衆国98052 ワシントン州レッ
 ドモンド ワンハンドレッド・アンド・フ
 イフティエイス・プレース ノースイース
 ト 8336

(72)発明者 パトリック・エム・ブランド
 アメリカ合衆国27613 ノースカロライナ
 州ローリー ウィローウッド・コート
 8904